AMENDMENTS TO THE CLAIMS

Listing of Claims in the case

The following listing of claims replaces all prior versions:

1- 36. (Cancelled)

37. (Currently Amended) A method of scheduling and executing instructions comprising:

a) accessing a sequence of instructions comprising:

a first memory operation that involves a first address range;

a second memory operation that involves at least a portion of said first address range; and

a third memory operation intervening said first and second memory operations, wherein it is not known whether said third memory operation involves an address within said first address range, wherein at least one of said first through third memory operations comprises a store operation;

b) eliminating said second memory operation from said sequence of instructions;

c) adding information to said third memory operation to allow determination of said first address range, wherein said information comprises a mask allowing determination of which of a plurality of registers hold protected addresses;

[[c)]] d) executing said sequence of instructions with said second memory operation eliminated; and

[[d)]] e) determining, during said executing, if said third memory operation involves an address within said first address range, and if so, raising an exception and re-executing the sequence of instructions including said second memory operation.

38. (Cancelled)

39. (Cancelled)

40. (Currently Amended) The method of Claim [[39]] 37, wherein said [[d)]] e) further comprises determining, during said executing, if said third memory operation involves an address within a range of any of said protected addresses.

41. (Currently Amended) The method of Claim [[39]] 37, further comprising storing a memory address associated with said first address range in one of said plurality of registers prior to said executing said sequence of instructions.

42. (Previously Presented) The method of Claim 37, further comprising storing a memory address associated with said first address range in a register prior to said executing said sequence of instructions.

43. (Currently Amended) The method of Claim 42, wherein:

said sequence of instructions comprises a fourth memory operation that is in said sequence of instructions after said first memory operation; and

further comprising adding second information to said fourth memory operation that allows said fourth memory operation to execute without exception even if said fourth memory operation involves said first address range.

44. (Previously Presented) The method of Claim 37, wherein said first and second memory operations would be safely reducible to a single memory operation if said third memory operation were not intervening.

45. (Currently Amended) A method of scheduling and executing instructions comprising:

a) accessing a sequence of instructions comprising:

a first load instruction that loads from a first address range;

a second load instruction that loads from said first address range; and

a store instruction intervening said first and second load instructions, wherein it is not known whether said store instruction stores to an address within said first address range;

b) eliminating said second load instruction from said sequence of instructions;

c) executing said sequence of instructions without said second load instruction, comprising storing a memory address associated with said first address range in a protection register; and

d) determining, during said execution, if said store instruction stores to an address within said first address range, and if so, raising an exception and re-executing the sequence of instructions including said second load instruction.

46. (Cancelled)

47. (Currently Amended) The method of Claim [[46]] 45, wherein said b) further comprises adding a flag to said store instruction to indicate said protection register.

48. (Previously Presented) The method of Claim 45, wherein said b) further comprises changing said first load instruction to a load and protect instruction.

49. (Currently Amended) A method of scheduling and executing instructions comprising:

a) accessing a sequence of instructions comprising:

a first store instruction to a first address range;

a second store instruction to said first address range; and

a load instruction intervening said first and second store instructions, wherein it is not known whether said load instruction involves said first address range;

b) eliminating said first store instruction from said sequence of instructions, comprising storing a memory address associated with said load instruction in a protection register;

c) executing said sequence of instructions with said first stored instruction removed; and

d) determining, during said executing, if said load instruction involves an address in said first address range, and if so, raising an exception and re-executing the sequence of instructions including said first store instruction.

50. (Cancelled)

51. (Currently Amended) The method of Claim [[50]] 49, wherein said b) further comprises adding a flag to said second store instruction to indicate said protection register.

52. (Previously Presented) The method of Claim 49, wherein said b) further comprises changing said load instruction to a load and protect instruction.

53-70. (Cancelled)

71. (Currently Amended) A method of scheduling and executing instructions comprising:

a) accessing a sequence of instructions comprising:

a first store instruction that stores to a first address range;

a load instruction that loads from said first address range; and

a second store instruction intervening said first store instruction and said load instruction, wherein it is not known whether said second store instruction stores to an address within said first address range;

b) eliminating said load instruction from said sequence of instructions, comprising storing a memory address associated with said first address range in a protection register;

c) executing said sequence of instructions without said load instruction; and

d) determining, during said execution, if said second store instruction stores to an address within said first address range, and if so, raising an exception and re-executing said sequence of instructions including said load instruction.

72. (Cancelled)

73. (Currently Amended) The method of Claim [[72]] 71, wherein said b) further comprises adding a flag to said second store instruction to indicate said protection register.

74. (Previously Presented) The method of Claim 71, wherein said b) further comprises changing said first store instruction to a store and protect instruction.

75. (Currently Amended) A method of scheduling and executing instructions comprising:

a) accessing a sequence of instructions comprising:

a load instruction that loads from a first address range;

a first store instruction, wherein it is not known whether said first store instruction stores to an address within said first address range;

a second store that stores to said first address range, wherein said first store instruction intervenes said load instruction and said second store instruction;

b) eliminating said second store instruction from said sequence of instructions, comprising storing a memory address associated with said first address range in a protection register;

c) executing said sequence of instructions without said second store instruction; and

d) determining, during said execution, if said first store instruction stores to an address within said first address range, and if so, raising and exception and re-executing said sequence of instructions including said second store instruction.

76. (Cancelled)

77. (Currently Amended) The method of Claim [[76]] 75, wherein said b) further comprises adding a flag to said first store instruction to indicate said protection register.

78. (Previously Presented) The method of Claim 75, wherein said b) further comprises changing said load instruction to a load and protect instruction.

79. (Previously Presented) The method of Claim 75, wherein said second store stores back to the first address range the same value that said load instruction loads from the first address range.